International Council on Systems Engineering (INCOSE)
7th European Systems Engineering Conference (EuSEC 2010)
Stockholm, Sweden, May 23rd – 26th 2010

# Conceptual Framework for Assessing and Measuring System Maturity, System Readiness and Capability Readiness

**Mr. Abideen Tetlay**
BA(Hons) MSc MSc(Oxon) FIAP
Software and Systems Engineer



## Abstract

This study summarises the issues with the notions of 'maturity' and 'readiness' within the context of the System Development and overall Lifecycle and provides formal definitions for these terms. It explains why a *Framework* is required to assess and measure Maturity and Readiness and how the conceptual Framework was devised, including a process for applying the Framework for the assessment and measurement of System Maturity, System Readiness and Capability Readiness.

**Keywords:** *System Maturity, System Readiness, Operational Capability, Capability Readiness, Measures of Effectiveness, Framework*

## Introduction

Today's systems are inherently complex due to a number of reasons, such as software and systems integration between subsystems, systems of systems and networked systems of systems. This level of complexity introduces a number of challenges both during the system development programme and the overall lifecycle. This level of complexity also increases the *risk* in system development, system integration and implementation which is often reflected in delays in system implementation and/or system failure, including systems showing undeterministic behaviour once released into the real world even though they were considered to be "ready" for use. The crucial objective of a lifecycle is to achieve a system that is "successful" in use in the real world and decisions made in the development of a system must be made with this aim and hence, the stakeholders must understand what "success" would be (from each stakeholder perspective) and what real world context and environment will the system be exposed to / interoperate with / be affected by and how would a potential system concept behave in such a context. These are issues that are often causes of problems in real world projects (Tetlay and John 2009b). During the development of a system, assessing the "maturity" of the system definition towards a successful outcome is important, as is the assessment of the "readiness" of a system to undertake roles within the real world (Tetlay and John 2009b). Therefore, we need to be able to assess and measure, with confidence, a System's Maturity and Readiness within a development programme and overall lifecycle (Tetlay and John 2010).

The aim of this study is to develop a theoretical Framework based on the research already undertaken by the authors (Tetlay and John 2010 and Tetlay and John 2009a,b) for the assessment and measurement of System Maturity, System Readiness and Capability Readiness.

This paper is structured as follows. First, we describe the methodology used for this study and summarise the issues with the notions of 'maturity' and 'readiness' within the context of the System Development and overall Lifecycle and then provide formal definitions for these terms using case study examples to illustrate the terms. This is followed by an explanation as to why a Framework is required to assess and measure Maturity and Readiness and how the conceptual Framework was devised, including a process for applying the Framework for the assessment and measurement of System Maturity, System Readiness and Capability Readiness in a development programme and overall lifecycle. Finally, conclusions are drawn and the next stages of our research are provided in terms of recommendations for further research.

## Methodology

The purpose of this section is to allow the reader to appreciate and assess the stringency and thoroughness with which the authors have sought out the evidence for this study. The authors have conducted their research for this study based on a 'scientific method' using a 'scientific attitude' (Robson 2002). This study was therefore carried out *systematically*, *sceptically* and *ethically*:

- *Systematically* means giving serious thought to what you are doing and how and why you are doing it; in particular, being explicit about the nature of the observations that are made, the circumstances in which they are made and the role you take in making them;
- *Sceptically* means subjecting your ideas to possible disconfirmation and also subjecting your observations and conclusions to scrutiny (by yourself initially, then by others); and
- *Ethically* means that you follow a code of conduct for the research which ensures that the interests and concerns of those taking part in, or possibly affected by the research are safeguarded (Robson 2002).

Such empirically based, systematic, sceptical and ethical research should, in the real world, also be in some sense influential or effective if it is to be worthwhile. Working in this way, systematically, being explicit about all aspects of your study and opening up what you have done to scrutiny by others is likely to lead to better-quality, more useful research (Robson 2002). This study falls within the scope of the overall research project as depicted in Figure 1.

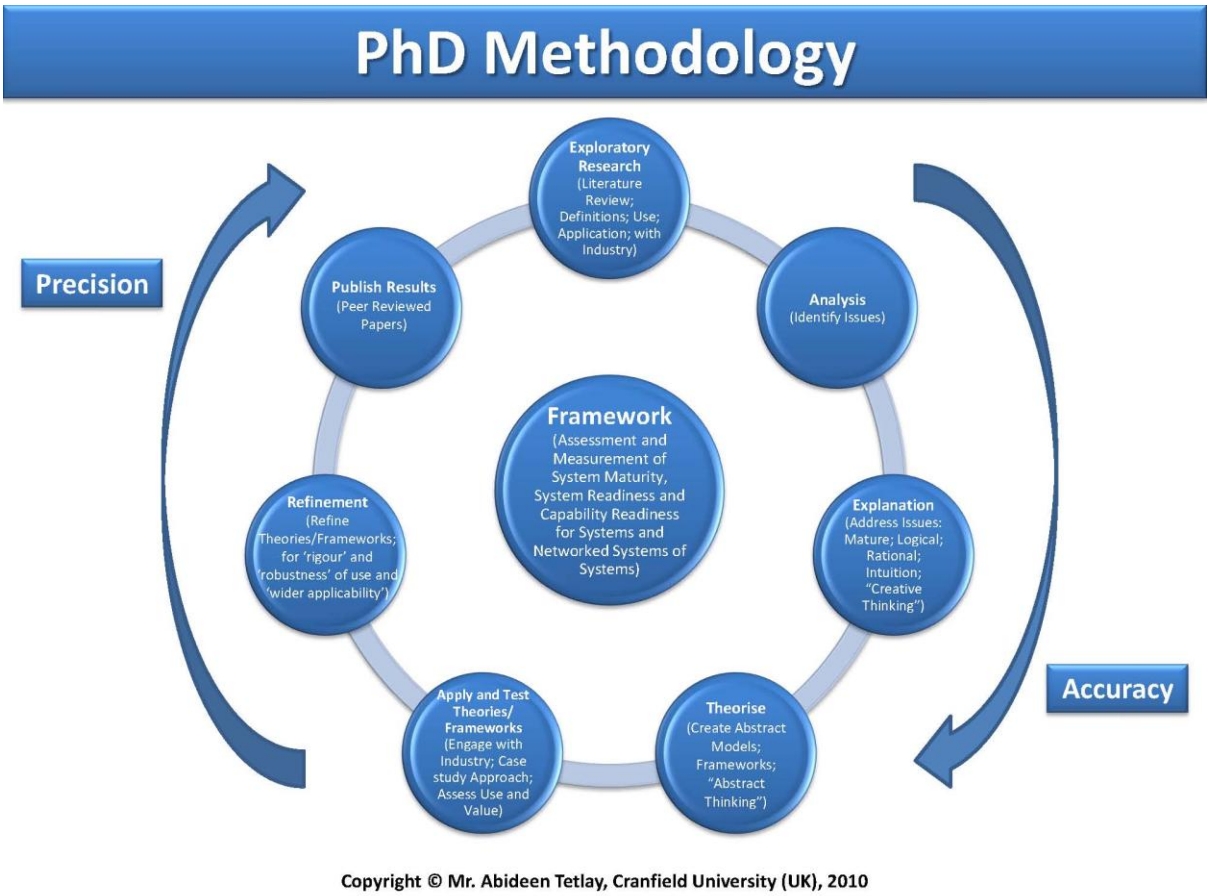


Figure 1: PhD Methodology

The first stage is of an exploratory nature where you explore and review the literature with an open-mind. This exploration phase involves reviewing the literature using key terms already identified from the initial background reading on the subject matter. The literature is reviewed to identify any previous work already undertaken in academia and industry looking for any gaps and issues in the literature. The literature is critically analysed and evaluated for the identification of definitions and how they are being used and applied in terms of their application; in theory and practice and the results published accordingly (Tetlay and John 2009b). In the next stage of the research, the issues are analysed and an explanation for the issues is determined. The intention is to address the issues based on a mature, logical, rational and intuitive approach 'creative thinking' leading to a conceptual theory/model and/or framework 'abstract thinking' and the results published accordingly (Tetlay and John 2010 and Tetlay and John 2009a,b). The theory and framework are then refined, applied and tested for validation, either empirically or in conjunction with industry in terms of a case study approach. Relevant secondary data can also be used as a case study assuming all parameters can be satisfactorily tested (Yin 2009). The intention is to assess the use and value of the framework, ideally in the real world with industry collaboration. This is part of our on-going research. Therefore, in the final stages of the overall research project, refinement of the theories and frameworks takes place for 'rigour' and 'robustness' of use and for wider applicability. At the end of each stage of the research project, the results of the research are then published, ideally first scrutinised by peer review for 'scientific rigour'. It is important to note that this research cycle is an iterative rather than a sequential process.

The Research Design used for this study is illustrated in Figure 2. The research proposal determines the aim, objectives and the research question, albeit, informally, especially at the beginning of the research and which may well change as you progress through your research and it becomes more mature in its definition towards the end of the research project.

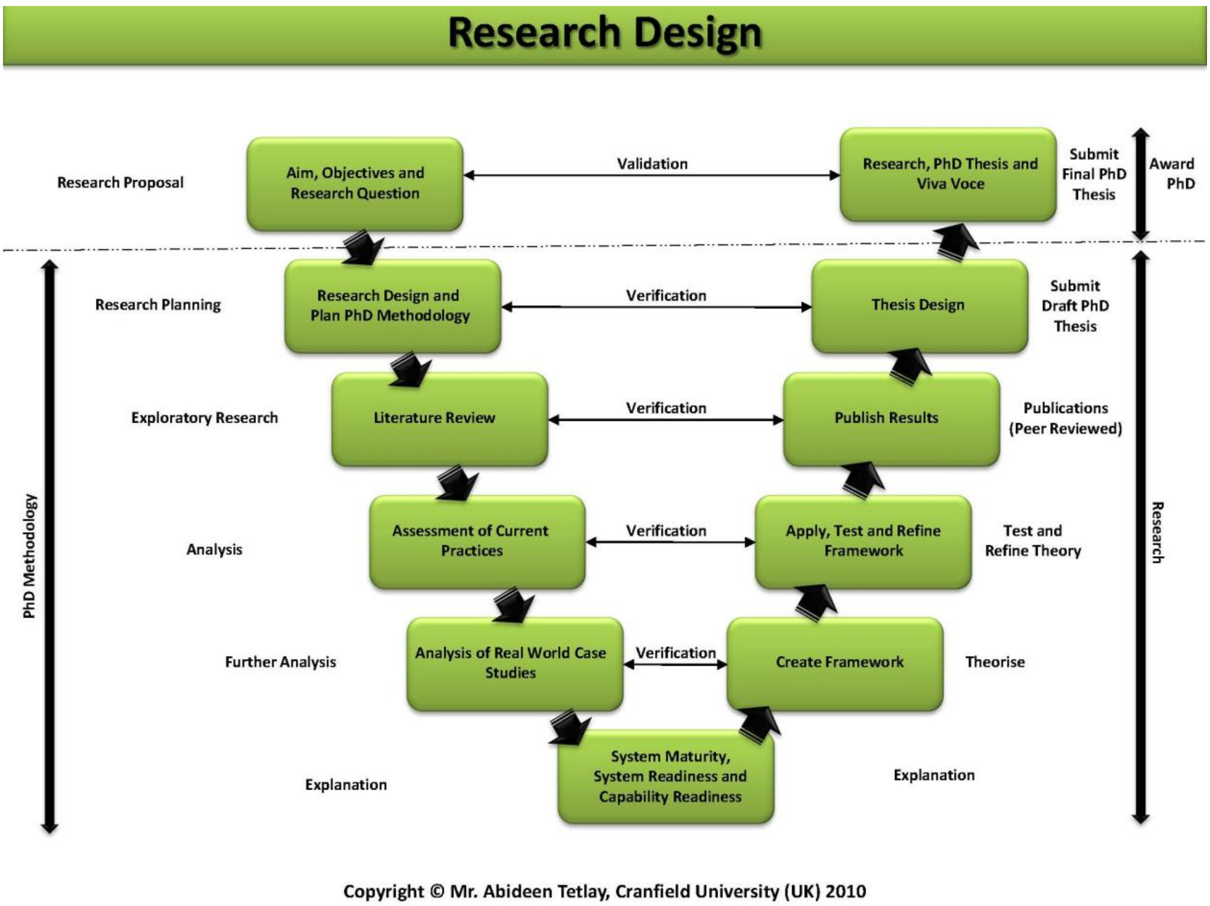


Figure 2: Research Design

The research planning phase of the research determines the type of research design to be implemented, either fixed design 'quantitative'; flexible design 'qualitative'; or mixed method 'qualitative' and 'quantitative'. Those following what some call 'qualitative' approaches (sometimes referred to as 'flexible' designs) *need* not shun being 'scientific' (Robson 2002). This study is based on a flexible design as shown in Figure 2. Since the early 1990s the design of flexible studies using qualitative methods has excited much interest and generated many publications. There are strong arguments for characterizing both fixed and flexible designs as scientific, provided that they are carried out in a systematic, principled fashion (Robson 2002). The research design for this systems engineering research project as shown in Figure 2 is depicted in the form of a V-Model mimicking the System Engineering Lifecycle. In terms of verification and validation, the left hand-side of the model is focusing on the planning and design stages of the research, including exploratory research and analysis identifying issues early on in the research project. The right hand-side focuses on creating the theory and framework and then applying and testing it for verification, through refinement, as appropriate, for wider use and applicability. Validation of the research is achieved once the research aim and objectives have been met and the research project submitted and defended accordingly. The aim of the overall research project is to define a suitable approach for the understanding and defining of System Maturity, System Readiness and Capability Readiness measures which will enable the development and utilisation of component systems of networks to be conducted with increased rigour and confidence and at reduced risk by making an assessment and measurement of Maturity and Readiness for systems and networked system of systems throughout a development programme and overall lifecycle.

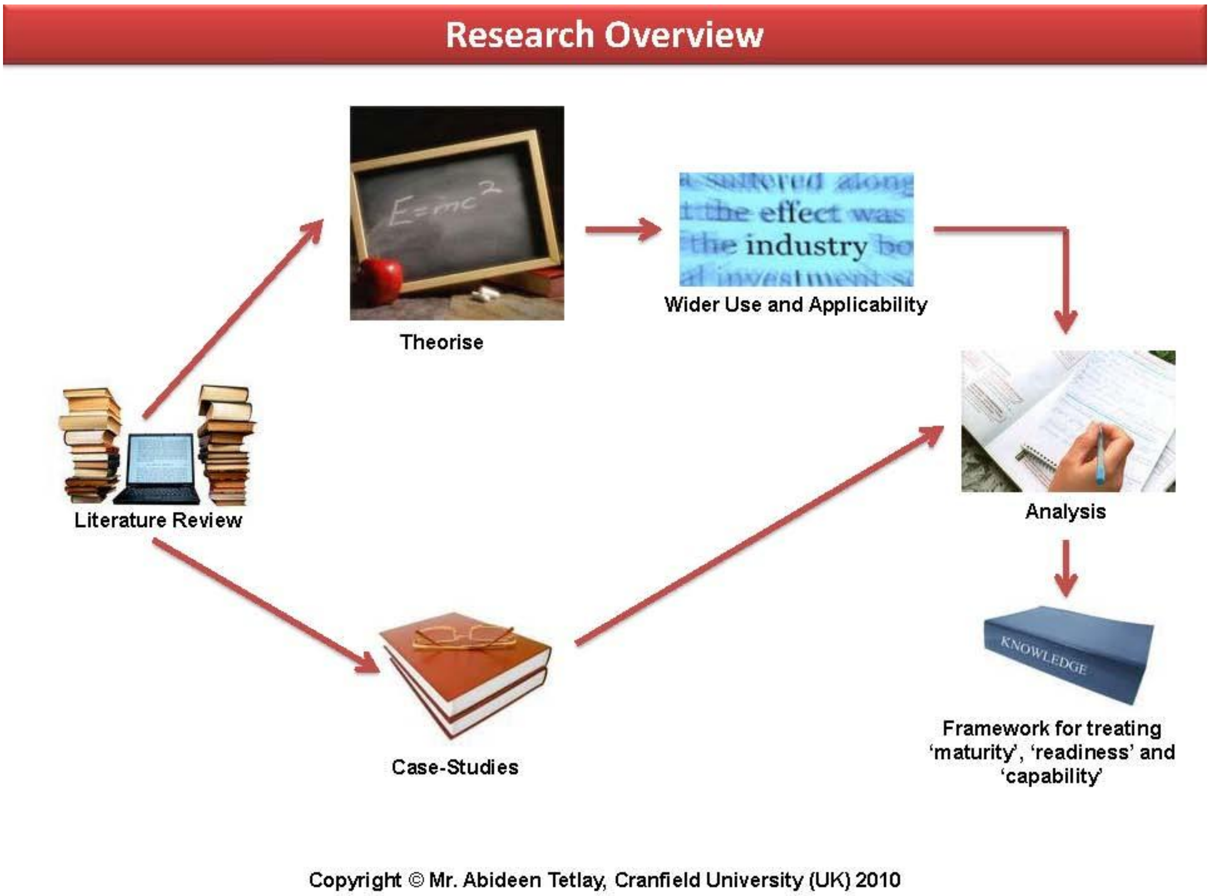


Figure 3: Research Overview

The objective of the overall research project is to establish a clear, useful Framework for assessing and measuring System Maturity, System Readiness and Capability Readiness, including the development of a rigorous "metric", a process for its use within a development programme and overall lifecycle and how it applies at different system levels, from individual subsystem to networked systems of systems. In terms of the progress made to date on the overall research project, the authors have already analysed the current literature and identified the issues accordingly (Tetlay and John 2009b). These are summarised in the next section of this paper. New terms and concepts have been created (Tetlay and John 2010 and Tetlay and John 2009b) and applied to a different domain for wider use and applicability (Tetlay and John 2009a). Clarification and refinement of existing and new terms and concepts has also been completed through case studies (Tetlay and John 2010). The case studies analysed in (Tetlay and John 2010) are also used for the basis of creating the conceptual Framework for treating 'maturity', 'readiness' and 'capability' as illustrated in Figure 3 and is the aim of this study.

The core methodology of this study is Case Studies. The authors selected this particular approach to challenge theoretical work previously undertaken in (Tetlay and John 2010 and Tetlay and John 2009a,b); to allow for the in-depth exploration of solutions for complex issues identified in (Tetlay and John 2010 and Tetlay and John 2009a,b) within its real-life context, especially when the boundaries between phenomenon and context are not clearly evident (Yin 2009); and benefits from the prior development of theoretical propositions in (Tetlay and

John 2010 and Tetlay and John 2009a,b) to guide data collection and analysis (Yin 2009). The authors chose multiple case studies to allow them to make analytic generalisation and not for statistical generalisation (Robson 2002). They hand-picked the following three case studies: Chinook Helicopter; Apache Helicopter; and the Type 45 Anti-Air Warfare Destroyer. These case studies were specifically identified by the authors as being relevant to their research, because they wanted to examine both new and existing defence systems spanning a significant length of time (50-60 years) developed by multiple defence contractors and sub-contractors across the US, UK and Europe. They specifically selected defence systems, because they are not looking at "maturity" or "capability" from a CMMISM (CMMISM 2008) process perspective as explained in (Tetlay and John 2010 and Tetlay and John 2009a,b), i.e. the process maturity of the capability, but in terms of the Operational Capability of the system or product, similar to the way the UK MoD looks at capability from a military operational capability perspective (MoD 2008a,b).

## System Maturity and System Readiness

In (Tetlay and John 2009b) the authors suggested that the notions of "maturity" and "readiness" are completely meaningful only within a contextual setting: "mature enough for what?" and "ready for what?" We need to be able to judge and express a system's maturity to assess when we have achieved a defined and implemented system. A meaningful view of the "maturity" of a system is pertinent to the determination of Risks associated with its development and operation. This notion applies at all system levels: for the integration of a system into a wider environment with other systems, for the design of a system itself and for any particular part of that system. We want to reach a situation where we are more confident that we shall not experience the unexpected and unacceptable problems we have often seen in the lifecycle (Tetlay and John 2009b).

However, in (Tetlay and John 2009b) the authors showed that there is not a sufficiently clear distinction between System Maturity and System Readiness. Current definitions of Readiness Levels seem to assess System Maturity in order to determine System Readiness and they all refer to the key term of "maturity" in their definitions. Based on this evidence, you could infer that the notion of "maturity" is encapsulated within the notion of "readiness" and they appear to be used interchangeably. It could also be argued that the existing Readiness Levels actually provide a "maturity" metric as opposed to a "readiness" metric. This is likely to lead to the confusion in the understanding of the terms of "maturity", i.e. System Maturity and "readiness", i.e. System Readiness in systems engineering and in the use of existing Readiness Levels (Tetlay and John 2009b).

To address these issues, in (Tetlay and John 2009b) the authors suggested that clarification is required for the terms of System Maturity and System Readiness because the potential use of these terms are very important. They are directly linked to Risk, Progress and Fitness for Purpose. The success of systems and system projects would benefit from a more meaningful and appropriate use of these concepts. If current levels of confusion continue it may have adverse consequences for the treatment of Risk, Progress and Fitness for Purpose (Tetlay and John 2009b).

In order to help clarify the notions of "maturity" (System Maturity) and "readiness" (System Readiness), in (Tetlay and John 2009b) the authors advocated that these terms and notions should be treated as two clear and distinct entities both of which are actually addressing two completely different *questions* within the System Development and overall Lifecycle. In (Tetlay and John 2010) the authors have further clarified the concepts of System Maturity and System Readiness using three case studies: Chinook Helicopter; Apache Helicopter; and the Type 45 Anti-Air Warfare Destroyer. This process resulted in the creation of a new set of System Maturity Levels to define System Maturity and a System Readiness Model to define System Readiness as depicted in Figure 4 and Figure 5, respectively.

The left hand-side of the model (see Figure 4) focuses on the 'Design Maturity' (consistency, completeness, coherence and confidence) for the system or product being engineered and the right hand-side concentrates on achieving verification, i.e. System Maturity. The purpose of the System Maturity Levels is to determine where you are in the System Development Lifecycle which determines the degree of System Maturity for the system or product currently being developed. The left hand-side of the System Development Lifecycle is less 'mature' than the right hand-side. Obviously, the further you are in the System Development Lifecycle, moving from the left to the right hand-side, the closer you are towards achieving a physical system or product and therefore achieving System Maturity (Tetlay and John 2010).

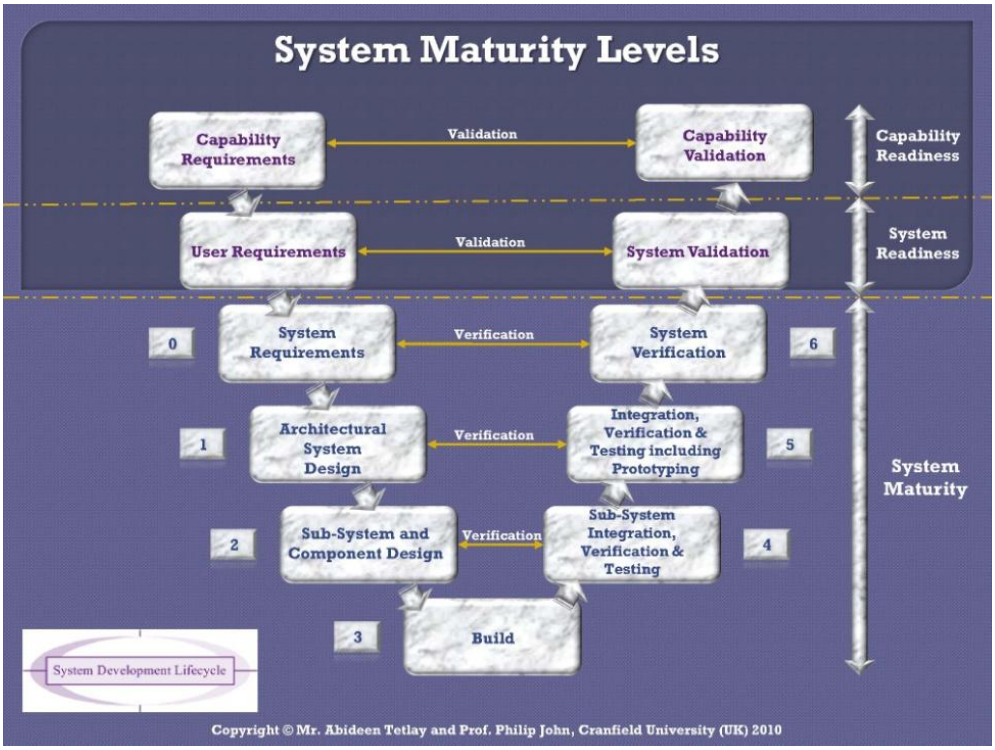


Figure 4: System Maturity Levels

The new revised definition for **System Maturity** which takes into account the new System Maturity Levels is provided below (Tetlay and John 2010):

> System Maturity is the **verification** within an **iterative process** of the System Development Lifecycle and occurs before System Readiness, i.e. the system or product must first be fully 'mature' before it can be "ready" for use. The process starts from **System Requirements** and finishes at **System Verification** (see Figure 4). System Maturity asks the *question*: Do we have a complete, well defined design that has been implemented and verified, i.e. we have decided what we want to implement and we have achieved this; the designed system or product now physically exists (the long horizontal dotted-line between User Requirements and System Requirements in Figure 4 denotes the physical existence of the produced-engineered system or product and also separates System Maturity from System Readiness)? System Maturity is both **solution dependent** and **context specific**. The System Requirements implicitly determines the context of use. System Maturity is only concerned with the **intrinsic aspects** of the system or product, i.e. the system or product features.
>
> Three phases or states of "**System Maturity**" could be envisaged:
> (1) System is Immature (SI) - In the System Requirements stage of the System Development Lifecycle or just completed it. System Maturity Level is equal to 0;
> (2) System Maturity is in Progress (SMP) - Working your way through the System Development Lifecycle (for example, the design, development and testing part of the System Development Lifecycle) in order to decide and define the system design and bring it into physical existence. This covers System Maturity Levels 1 to 5, inclusive; and
> (3) System Maturity has been Achieved (SMA) - The design, development and testing of the system or product is now complete, fully "mature" and tested and now physically exists. To achieve System Maturity the system or product must be verified against the System Requirements, i.e. you have achieved System Maturity by building the system right based on 'Best Practice' procedures and 'Standards' in place. This can only occur if all the Aspects of System Maturity (AoSM) are in place, i.e. the stages of the System Development Lifecycle (see Figure 4). System Maturity Level is equal to 6 (Tetlay and John 2010). Verification confirms that the system element meets the design-to or build-to specifications. It answers the *question* "Did you build it right? (CMMISM 2008)".

In (Tetlay and John 2010) the authors also provided some examples of System Maturity to further explain and clarify this important concept. The following are all examples of systems or products failing to achieve System

Maturity, because they have failed to build the systems or products based on **'best practice' procedures** and/or **'standards' in place**. They also illustrate the importance of first achieving System Maturity before you can achieve System Readiness, i.e. before the system or product can even be considered to be "ready" for use which is the 'Fitness for Purpose' question:

- "The Chinook Mk3 helicopters have not flown on operations because the Department refused to grant the helicopters an **airworthiness certificate**. Although Boeing met its contractual obligations, the avionics software fell short of United Kingdom military **airworthiness standards** and the helicopters have not flown on operations" (NAO 2008);
- "By the time the airworthiness issues with the Chinook Mk3 helicopters became apparent, **emerging legislation** and operational requirements meant that the helicopters would need to be modified before they could be deployed on operations" (NAO 2008);
- "The Reversion project aims to deliver the eight Chinook Mk3 helicopters to a **standard** such that they can be deployable to Afghanistan in 2009-10" (NAO 2008);
- "The US Secretary of State requires all aircraft, associated equipment and software to be designed, constructed, maintained and operated in such a manner that they are airworthy" (NAO 2008); and
- "There is still some risk to the delivery of the Apache as development work to install a range of more recently contracted enhancements to the baseline helicopter has yet to be completed" (NAO 2002).

The examples below demonstrate System Maturity as being **solution dependent**:

- "In September 2004 the Department decided to explore further a **"Fix to Field" solution** which involved the replacement or modification of the cockpit display systems, the communications systems and fuel quantity gauging, together with the integration of special operations equipment and a comprehensive Defensive Aids Suite" (NAO 2008); and
- "Each confirmed that the **solution** was technically feasible, had a sound approach to airworthiness and would enable the Chinook Mk3 to be declared an operationally effective helicopter to be used on special operations" (NAO 2008).

The following example shows that System Maturity is only concerned with the **intrinsic aspects** of the system or product, i.e. the system or product features:

- "Three Chinook airframes, CH-47A, CH-47B, and a CH-47C, were stripped down to their basic airframes and then rebuilt with improved systems to provide three CH-47D prototypes. Improvements included upgraded power plants, rotor transmissions, integral lubrication and cooling for the transmission systems, and fiberglass rotor blades. Other improvements included a redesigned cockpit to reduce pilot workload, redundant and improved electrical systems, modularized hydraulic systems, an advanced flight control system, and improved avionics" (FAS 2008).

In order to achieve **System Maturity** a system or product must be verified against the **System Requirements**. The following are examples of **System Requirements**:

- "The fuel cells must be crash-worthy and self sealing up to 50 caliber hits" (FAS 2008); and
- "Type 45 shall carry a Medium Calibre Gun System of at least 114mm" (NAO 2009a,b).

The following is an example of **Design** which is **Immature** leading to increased risk and uncertainty:

- "The Department and its commercial partners were over-optimistic in their predictions of the time and resources required to procure the first six ships, and did not establish the project on a suitable basis given the levels of risk and uncertainty and the immaturity of the design of the ships and the PAAMS missile system" (NAO 2009a,b).

However, the following is an example of **Design** which is **Mature**. Note that the design is able to cope with new requirements, including Urgent Operational Requirements (UOR) and requirements which have yet to be determined:

- "The Type 45 could also accommodate cruise missiles such as the Tomahawk and anti-ballistic missiles if a requirement was identified in future" (Net Resources International 2009b).

The new revised definition for **System Readiness** is provided below (Tetlay and John 2010):

> System Readiness is the **validation** aspect of the system development and overall lifecycle and occurs after System Maturity, i.e. the system or product must first be fully 'mature' before it can be made "ready" for use. The process starts from **User Requirements** and finishes at **System Validation** (see Figure 4). System Readiness determines whether or not the system or product is now "ready" for use in its intended **operational environment**. System Readiness is a relative

metric based on context and use, i.e. the **'Fitness for Purpose'** question. System Readiness is **context dependent**. The User Requirements explicitly determine the context of use. System Readiness is **solution independent**, i.e. users do not care about the solution, only whether or not the system or product is "ready" for use. System Readiness is only concerned with the **extrinsic aspects** of the system or product, i.e. is the system or product now "ready" to be used in the real-world context? System Readiness is dependent on **Enablers** and **Barriers**. Measures of Effectiveness (MoE) can be used to assess and measure the **system's or product's effectiveness**. To achieve System Readiness the system or product must be validated against the User Requirements, i.e. you will achieve System Readiness by building the right system or product for a given context.

Three phases or states of "**System Readiness**" could be envisaged:
(1) No System Readiness (NSR) – Certain enablers for the system or product for a particular context are not currently in place and certain barriers are also preventing the system or product from being operational and "ready" for use; conceptually, this can be thought of as System Readiness being equal to 0;
(2) Initial System Readiness (ISR) - Certain enablers for the system or product for a particular context are currently in place, but certain barriers are preventing the system or product from being *fully* operational and therefore the system or product only has limited operational use for a particular context, but is "ready" for use for that context only; conceptually, this can be thought of as System Readiness being equal to 1; and
(3) Full System Readiness (FSR) – All the enablers for the system or product for a particular context are currently in place and none of the current barriers are preventing the system or product from being *fully* operational and the system or product has *full* operational use for a particular context and is "ready" for use for that context only; conceptually, this can be thought of as System Readiness being equal to 2 (Tetlay and John 2010). Validation answers the *question* of "Did you build the right thing? (CMMISM 2008)". Note that this question is implicitly context dependent, i.e. 'right for what'?

In (Tetlay and John 2010) the authors created a conceptual model entitled, "System Readiness Model" as depicted in Figure 5 to try and capture the notion of 'maturity' (System Maturity) and 'readiness' (System Readiness). As you can see in Figure 5, we have taken the system development stages of the System Development and overall Lifecycle and labelled these as Aspects of System Maturity (AoSM) which need to be in place to achieve System Maturity and we have labelled the left hand-side of the System Maturity part of the model as 'Design Maturity' and the right hand-side as 'System Maturity'.

Sitting above System Maturity is System Readiness showing all the component parts of System Readiness. In order to determine System Readiness of a system or product that depends on whether or not the system or product can be 'validated' against the User Requirements and is dependent on the Enablers and Barriers as well as the context of use (Tetlay and John 2010). Measures of Effectiveness (MoE) could be used to assess and measure the degree of System Effectiveness (Verma et al., 2003).

In (Tetlay and John 2010) the authors also provided some examples of System Readiness to further explain and clarify this important concept. The following are all examples of System Readiness, because they are concerned with how the system or product is likely to behave in its intended **operational environment** for a given **context**:

- "...speed for any mission will vary greatly depending on load configuration (internal or external), time of day, or weather conditions" (FAS 2008);
- "The amount of load a cargo helicopter can carry depends on the model, the fuel on board, the distance to be flown, and atmospheric conditions" (FAS 2008);
- "The Chinook's primary mission is moving artillery, ammunition, personnel, and supplies on the battlefield. It also performs rescue, aeromedical, parachuting, aircraft recovery and special operations missions" (FAS 2008);
- "Chinooks (CH-47D) can fly more than 150 mph at full load more than 330 nautical miles with longrange fuel tanks. With a crew of three, the CH47s can transport 44 seated troops or 24 casualty litters" (Boeing 2008a,b); and
- "Daring will be able to transit 7,000 n miles at a speed of 18 kt and reach a maximum speed of more than 27 kt" (NAO 2002).

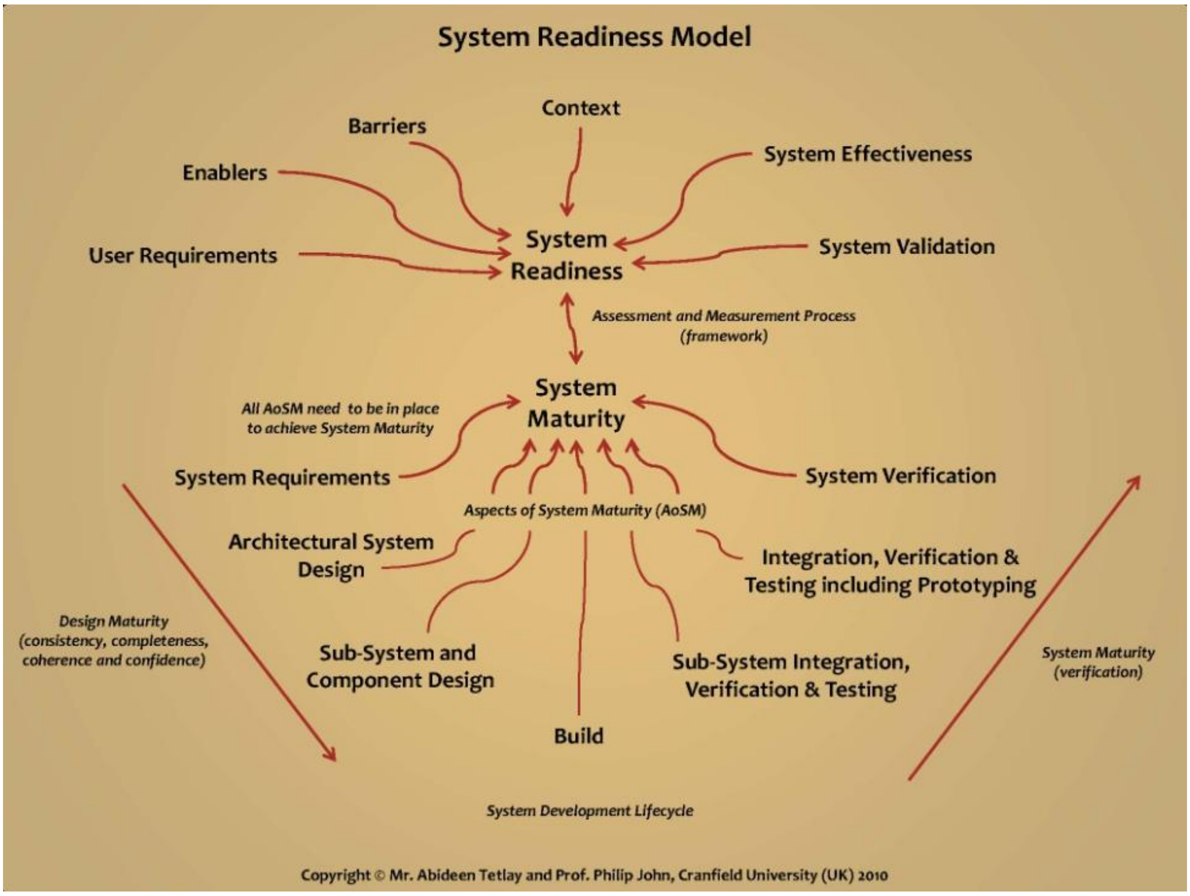


Figure 5: System Readiness Model

The following are examples of System Readiness, but in particular, **Initial System Readiness (ISR)** due to certain **Enablers** and/or certain **Barriers** in existence:

- "The helicopters can fly but are restricted to flying on cloudless days above 500 feet where the pilot can navigate via landmarks" (NAO 2008);
- "...the limitation aspects of operating the aircraft that are applicable to the aircraft systems, rather than flying the aircraft. Thus it covers items such as: Engine, Fuel System, Electrical System, Autopilot, Communications, and Radar" (NAO 2008);
- "Without their primary weapons systems these Type 42s could not be deployed in an air defence role". (NAO 2009a,b); and
- "...the Army determined that aircraft (Chinook CH-47D) with certain gear and bearing assemblies can continue training and perform operational missions that are limited to 80 percent, dual-engine torque" (FAS 1999).

The following example illustrates the importance of **User Requirements** in achieving System Readiness, i.e. in order to achieve System Readiness a system or product needs to be validated against the User Requirements:

- "When it is in service, each Type 45 destroyer should have met all of its key user requirements" (NAO 2009a,b); and
- "On current plans, Daring, the first of class will enter service in December 2009, when it will meet all its defined Key User Requirements... (NAO 2009a,b).

The following is an example of a **User Requirement**:

- "Type 45 shall be able to provide close tactical control to at least four, fixed wing or groups of, aircraft" (NAO 2009a,b).

The following is an excellent example of where a system or product has achieved System Readiness, but not necessarily Capability Readiness which we will discuss in the next section:

- "On current plans, Daring, the first of class will enter service in December 2009, when it will meet all its defined Key User Requirements. There are, however, a number of risks both to achieving these dates and to delivering the full capability in the longer term. In the short term these include integrating the destroyer and PAAMS; and trialling and operating the Combat Management System" (NAO 2009a,b).

## Capability Readiness

Many systems, whether new or existing, still continue to cause unexpected and unacceptable behaviour even though they were considered to be "ready" for use. Clearly, we are not proficient enough at understanding how systems should behave in the real world context and the "capability" expected of them (Tetlay and John 2010). In (Tetlay and John 2009a) the authors suggested that just because a system or product is now considered to be "ready" for use in its intended operational environment, does not necessarily mean that the system's or product's "capability" is also ready. Therefore, the concept of "Readiness", i.e. System Readiness may be further expanded and related to "capability", i.e. Capability Readiness. In (Tetlay and John 2009b), the authors first coined the term "Capability Readiness" and then extended the definition in (Tetlay and John 2009a). The authors mapped 'Capability Requirements' at the very beginning of the V-Model as illustrated in Figure 4 and before User Requirements which has traditionally been the starting point of the model. The premise for this is to ensure that you capture the *full* "complete" requirements starting from and including the 'Capability Requirements' which you need to build and factor into the System Development and overall Lifecycle (Tetlay and John 2009a,b). In (Tetlay and John 2010) the authors have further clarified the concept of Capability Readiness using three case studies: Chinook Helicopter; Apache Helicopter; and the Type 45 Anti-Air Warfare Destroyer. This process resulted in the creation of a new Capability Readiness Model to define Capability Readiness as depicted in Figure 7.

The new revised definition for **Capability Readiness** is provided below (Tetlay and John 2010):

> Capability Readiness determines whether or not the **'total-wider system or product'**, including Systems of Systems (SoS) and Networked Systems of Systems, for example, Networked Enabled Capability (NEC) has the **ability** and the **capacity** to completely fulfil the **operational capability** of the system or product for a given **context** in its intended **operational environment**, within the **scope** of the **Capability Requirements** and its aims and objectives. Once we know that the system or product has achieved System Readiness then we can raise the Capability Readiness question. Like System Readiness, Capability Readiness is also looking at the **validation** of the system or product and is also **context dependent**. The process starts at **Capability Requirements** and finishes at **Capability Validation** (see Figure 4). The Capability Requirements explicitly determine the context of use. To achieve Capability Readiness the system or product must be **validated** against the Capability Requirements, i.e. you will achieve Capability Readiness if you can "demonstrate" that the system or product does have the **ability** and the **capacity** to completely fulfil the **operational capability** of the system or product for its intended **operational environment** as prescribed by the Capability Requirements. Capability Readiness is a **relative metric** based on context and use, i.e. the **'Fitness for Purpose'** question. Capability Readiness is **solution independent**, i.e. users do not care about the solution, only whether or not the system's or product's **capability** is "ready" for use. Capability Readiness is only concerned with the **extrinsic aspects** of the system or product, i.e. is the system's or product's **capability** now "ready" to be used in the real-world context? Capability Readiness is dependent on **Enablers** and **Barriers**. Measures of Effectiveness (MoE) can be used to assess and measure the system's or product's **operational capability effectiveness**. Capability therefore translates to the system's or product's **"Ability"** (actual/physical behaviour) and **"Capacity"** (assumed/predicted 'potential' behaviour) to achieve a need as depicted in Figure 6 below. In terms of the assessment and measurement of Capability Readiness, this consists of three things: the 'assumed' (based on assumptions, conjecture and anecdotal evidence), 'predicted' (based on historical evidence; development of a prototype/simulation for example) and 'actual' (based on current, physical/real-life and real-time) result of the behavioural aspects of a system or product using both qualitative (non-formal methods) and quantitative (formal methods) techniques (Tetlay and John 2009a). When looking at Capability Readiness, it is important to recognize the **'total-wider system or product'** and to distinguish this from the 'produced-engineered system or product' which is composed of conventional development as illustrated in Figure 4 above (Tetlay and John 2009a).
>
> This is a key distinction between 'Capability Readiness' and 'System Readiness'. System Readiness is only concerned with the 'produced-engineered system or product', whereas Capability Readiness is concerned with the '**total-wider system or product'** (the long horizontal dotted-line at the very top of the diagram in Figure 4 above between Capability Requirements and User Requirements separates and distinguishes System Readiness from Capability Readiness) (Tetlay and John 2009a).

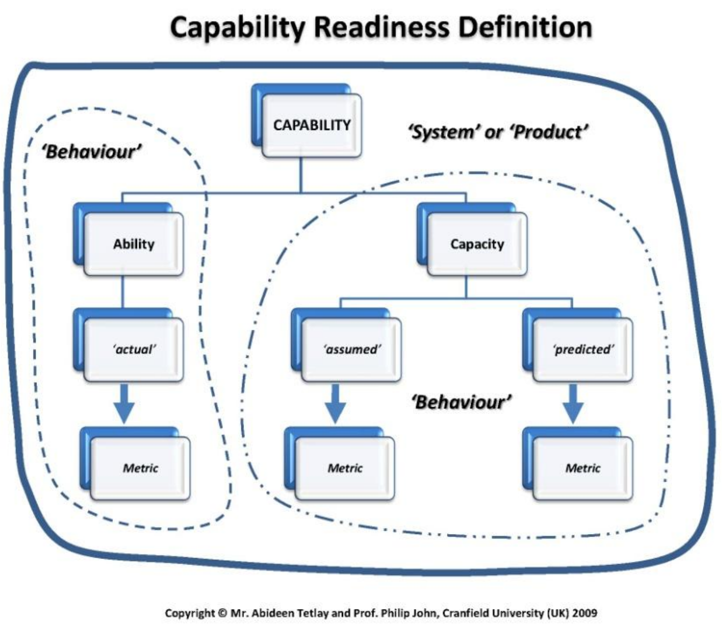


Figure 6: Capability Readiness Definition

Three phases or states of "**Capability Readiness**" could be envisaged:
(1) No Capability Readiness (NCR) – Certain enablers for the system or product for a particular context are not currently in place and certain barriers are also preventing the system's or product's **capability** from being operational and "ready" for use; conceptually, this can be thought of as Capability Readiness being equal to 0;
(2) Initial Capability Readiness (ICR) - Certain enablers for the system or product for a particular context are currently in place, but not all of them and certain barriers are preventing the system's or product's **capability** from being *fully* operational and therefore the system or product only has **limited operational capability** for a particular context, but is "ready" for use for that context only; conceptually, this can be thought of as Capability Readiness being equal to 1; and
(3) Full Capability Readiness (FCR) – All the enablers for the system or product for a particular context are currently in place and none of the current barriers are preventing the system's or product's **capability** from being *fully* operational and the system or product has **full operational capability** for a particular context and is "ready" for use for that context only; conceptually, this can be thought of as Capability Readiness being equal to 2.

In (Tetlay and John 2010) the authors created a conceptual model entitled, "Capability Readiness Model" as depicted in Figure 7 below to try and capture the notion of 'capability' (Capability Readiness). As you can see in Figure 7, the authors have taken the Defence Lines of Development (DLoD) (MoD 2008a,b) as a basis of their model. All the DLoD need to be in place to achieve Operational Capability, but in order to determine the Capability Readiness of a system or product that depends on whether or not the system or product can be 'validated' against the Capability Requirements and is dependent on the **Enablers** and **Barriers** as well as the context of use. Measures of Effectiveness (MoE) could be used to assess and measure the degree of Operational Capability (Verma et al., 2003).

In (Tetlay and John 2010) the authors also provided some examples of Capability Readiness to further explain and clarify this important concept. The following are all examples of Capability Readiness, because they are concerned with how the system's or product's capability is likely to behave in their intended **operational environment** for a given **context**:

- "Assuming an appropriate level of pilots and logistic support, additional Chinook flying hours could have been used to carry out operations with greater flexibility including more non-essential military tasks in support of the international mission in Afghanistan" (NAO 2008);
- "Past experiences of operating in a desert environment have resulted in restricted operations and reduced engine life". (FAS 1997); and
- "The new Type 45 Carriers and its aircraft are planned to be capable of operating in all weathers, day and night, flying strike missions, conducting offensive support for ground forces ashore and where necessary,

providing force protection to the fleet. The Aircraft Carrier will also be capable of supporting the operation of helicopters in a wide variety of roles including land attack and ground support" (NAO 2009a,b).

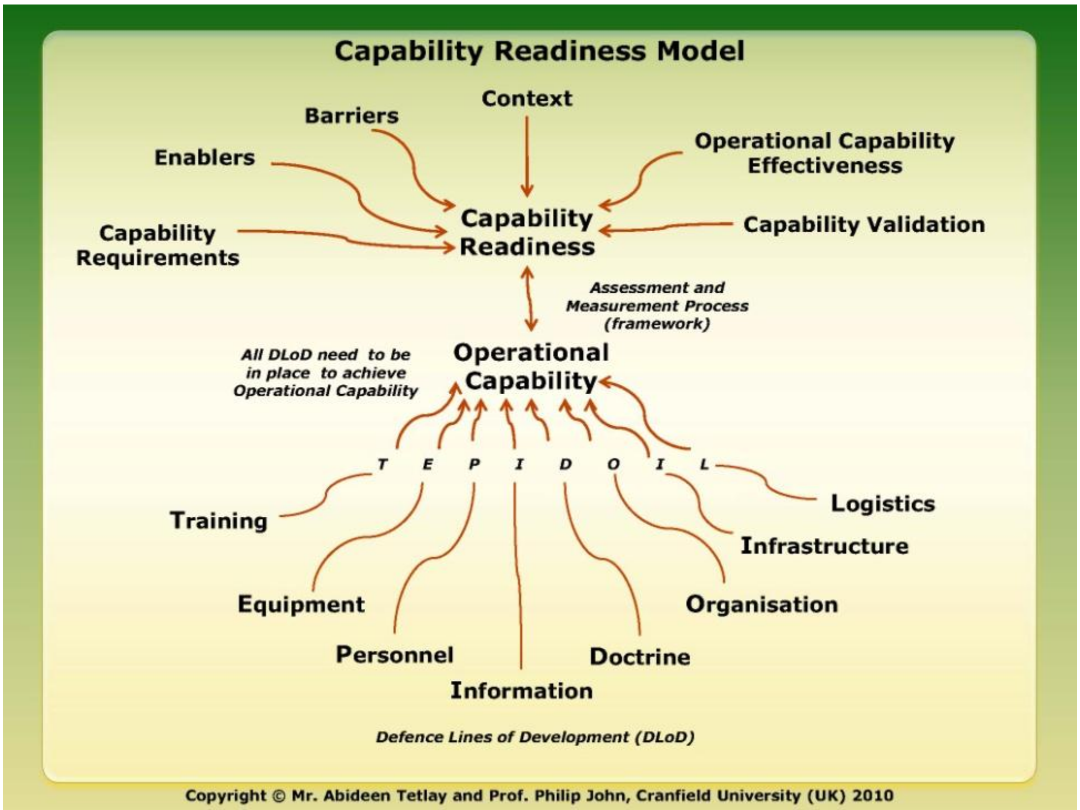


Figure 7: Capability Readiness Model

The following examples focus on the **DLoD** and their importance in providing **operational capability**:

- "...risks remain to be overcome on a number of Lines of Development before the Apache can be successfully introduced into service" (NAO 2002);
- "...the infrastructure should be in place for delivery of the initial capability" (NAO 2002);
- "...the delivery of Apache training has been delayed which has in turn delayed introduction of the capability" (NAO 2002);
- "The Department is bringing in the capability taking into account the Defence Lines of Development. This approach aims to ensure that all elements of the capability such as the necessary training and infrastructure, not just the ships themselves, are introduced coherently" (NAO 2009a,b); and
- "There are, however, some issues with the first four of the Defence Lines of Development that may limit the deployable capability if the Department's mitigating actions are not successful" (NAO 2009a,b).

The following are examples of Capability Readiness, but in particular, **Initial Capability Readiness (ICR)** due to lack of certain **Enablers** and current **Barriers** preventing full operational capability:

- "The hot mountainous conditions of Vietnam limited the Ch-47A models performance capabilities and generated a requirement for increased payload and better performance" (FAS 2008);
- "Clearance of the aircraft to operate in conditions of ice is targeted for December 2006" (NAO 2002);
- "...clearance for the Apache to operate in conditions of snow is planned for August 2003 and to operate in ice conditions is targeted for December 2006. This timescale means there will initially be some restrictions on the environments in which the aircraft can operate when it is introduced into service in 2004" (NAO 2002);
- "Because of the limited capability of other equipment the Apache will not, however, have the capacity for secure voice communications with the United Kingdom's Gazelle, Lynx, Sea King Mark 4 and Puma helicopters, nor will it be able to exchange data securely with most of the United Kingdom's military aircraft or the other battlefield helicopters or with United Kingdom ground forces" (NAO 2002);
- "Apache and Merlin Helicopters were deployed on operations before reaching full operating capability. Both have proved to be successful on those operations, Apache in Afghanistan, and the RAF version of Merlin in Iraq" (Slade 2008);

- "The delays on the Type 45 destroyer project mean that the Department is still actively operating five Type 42 destroyers which offer a much more limited capability" (NAO 2009a,b); and
- "...several pieces of equipment will be fitted to the destroyers incrementally after they come into service meaning that the full capability will not be available until the middle of the next decade" (NAO 2009a,b).

The following example is of Capability Readiness, but in particular, **Full Capability Readiness (FCR)**:

- "The full capability - a true integrated and joint capability that would allow the Apache to be based and supported at sea with an Amphibious Task Force for extended periods and in more demanding sea conditions - remains the final objective" (NAO 2002).

The following example illustrates the importance of **Capability Requirements** in achieving Capability Readiness, i.e. in order to achieve Capability Readiness a system or product needs to be validated against the Capability Requirements:

- "Deliver agreed capability against approved requirements" (NAO 2002).

The following are examples of **Capability Requirements**:

- "The Department is aiming to deliver at this point a capability that will allow the Apache to be re-fuelled and re-armed at sea and then fly to operations on land" (NAO 2002);
- "...escort, ground suppression and armoured reconnaissance. ...operate from ships of the Royal Navy and Royal Fleet Auxiliary" (Slade 2008); and
- "...be able to operate Lynx, Merlin and Chinook helicopters. ...to operate in a hostile environment, either to provide a protective umbrella over a force of landing ships, an aircraft carrier or a group of merchant ships, or to conduct a wide range of other tasks such as maintaining a United Kingdom presence, embargoes or supporting forces ashore" (NAO 2009a,b).

We have summarised the key characteristics of System Maturity, System Readiness and Capability Readiness as depicted in Table 1:

Table 1: Characteristics of System Maturity, System Readiness and Capability Readiness

**Characteristics of System Maturity, System Readiness and Capability Readiness**

| Concept | Scope | Solution | Operational Context | System Development Lifecycle | Verification and Validation (V-V) | Measures of Effectiveness (MoE) Perspective |
|---|---|---|---|---|---|---|
| Capability Readiness | Operational Capability (for the total-wider system or product, e.g. Product-Service Systems (PSS); System of Systems; Networked Systems of Systems, e.g. Networked Enabled Capability (NEC)) | Independent (user perspective – extrinsic aspects of the system or product) | Explicitly Context Dependent (user perspective) | Capability Requirements | Capability Validation ('fitness-for-purpose') | Operational Effectiveness (the effect of the Capability Requirement for a given context, user perspective) |
| System Readiness | User Functionality (for the produced-engineered system or product) | Independent (user perspective – extrinsic aspects of the system or product) | Explicitly Context Dependent (user perspective, e.g. CONEMP) | User Requirements | System Validation ('fitness-for-purpose') | System Effectiveness (the effect of the User Requirement for a given context, user perspective) |
| System Maturity | Technical Solution (trying to bring a design concept into physical existence) | Dependent (SE perspective - intrinsic aspects of the system or product) | Implicitly Context Dependent (SE perspective - there is an underlying context, e.g. CONUSE) | System Requirements | System Verification (based on 'best practice procedures' and 'standards' in place) | Technical Effectiveness (the effect of the System Requirement for a given context, e.g. sub-system integration and interoperability, SE perspective) |



## Conceptual Framework

This section explains why a conceptual *Framework* is required to assess and measure Maturity and Readiness and how the conceptual Framework was devised, including a process for applying the conceptual Framework for the assessment and measurement of System Maturity, System Readiness and Capability Readiness. The reason why we need to assess and measure Maturity and Readiness in the system development and overall lifecycle is extremely important. We need to be able to judge and express a system's maturity to assess when we have achieved a defined and implemented system. A meaningful view of the "maturity" of a system is pertinent to the determination of Risks associated with its development and operation. We need to judge "readiness" to assess when a system is 'Fit for Purpose' for a particular context (Tetlay and John 2009b). In (Tetlay and John 2009a), the authors discussed the relationship between "capability" and 'Product-Service Systems' (PSS) and the need for the assessment of Capability Readiness for PSS. They suggested that this assessment is essential to determine whether or not the elements of capability for PSS are in place and maintained for the successful delivery of a sustainable PSS. Novel approaches are required for evaluating the progress of decisions towards a successful "capability" operating in the real world (Tetlay and John 2009a,b). We want to reach a situation where we are more confident that we shall not experience the unexpected and unacceptable problems we have often seen in the lifecycle (Tetlay and John 2009a,b). Therefore, the aim of the conceptual Framework is to provide a formal (by procedure) conceptual Framework that stimulates thinking and drives a structured approach for the treatment of Maturity and Readiness, including a process for its use within a development programme and overall lifecycle which is directly linked to the reduction of risk.

The conceptual Framework described below has been derived from earlier work undertaken by (Tetlay and John 2010 and Tetlay and John 2009a,b). In (Tetlay and John 2010) the authors introduced a new set of System Maturity Levels and a conceptual model for System Readiness and Capability Readiness (as prescribed above), which we will use as a basis for the conceptual Framework along with the findings from the case studies. The case studies used in (Tetlay and John 2010) were the Chinook Helicopter, Apache Helicopter and the Type 45 Anti-Air Warfare Destroyer. Therefore, the conceptual Framework for the assessment and measurement of System Maturity is presented in Figure 8. We have used the new set of System Maturity Levels (Tetlay and John 2010) to create this conceptual Framework. The left hand-side of the conceptual Framework focuses on the Design and Development (System Maturity Levels 0 to 3, inclusive) for the system or product being engineered and the right hand-side concentrates on achieving verification (System Maturity Levels 4 to 6, inclusive), i.e. System Maturity (Tetlay and John 2010). The purpose of the conceptual Framework is to determine where you are in the System Development Lifecycle which determines the degree of System Maturity for the system or product currently being developed. The left hand-side of the System Development Lifecycle is less 'mature' than the right hand-side. Obviously, the further you are in the System Development Lifecycle, moving from the left to the right hand-side, the closer you are towards achieving a physical system or product and therefore achieving System Maturity (Tetlay and John 2010).

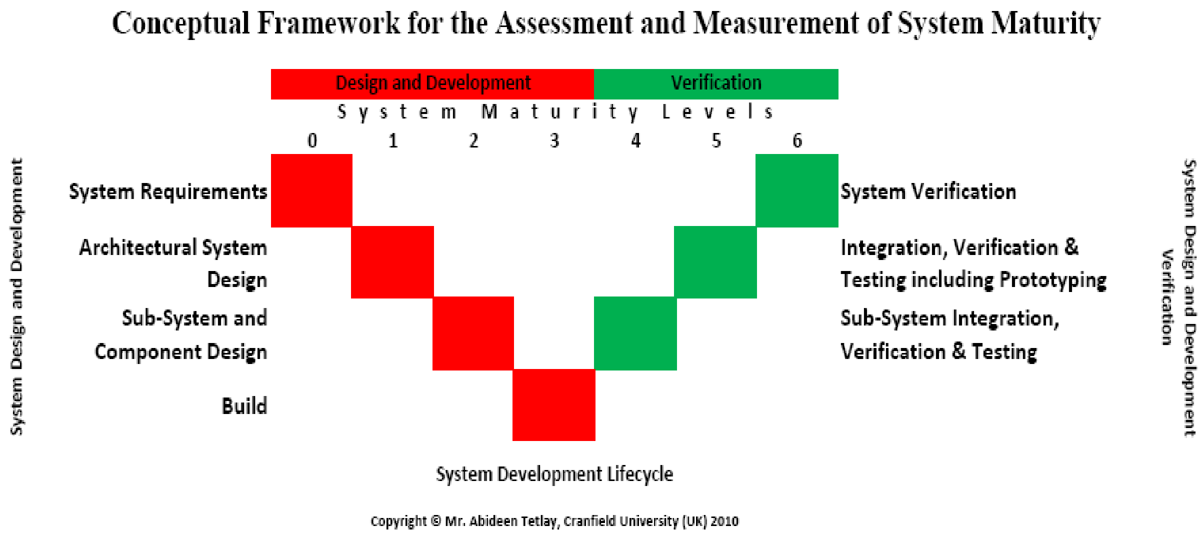


Figure 8: Conceptual Framework for the Assessment and Measurement of System Maturity

The procedure for the assessment and measurement of System Maturity is depicted in Figure 9. The main aim of this procedure is to *verify* the System Requirements of a system or product being engineered in order to achieve System Maturity. The procedure should be used by system engineers and project managers to help monitor and communicate the progress of a system engineering project by assessing and measuring its System Maturity, feeding into the project planning and to enable the Systems Design Authority and Systems IPT Lead to identify and address risks and mitigating actions in a consistent manner with confidence.

The conceptual Framework for the assessment and measurement of System Readiness is illustrated in Figure 10. System Readiness is concerned with the extrinsic aspects of the produced-engineered system with respect to how the system is expected to behave in a particular context subject to certain enablers and barriers in place.

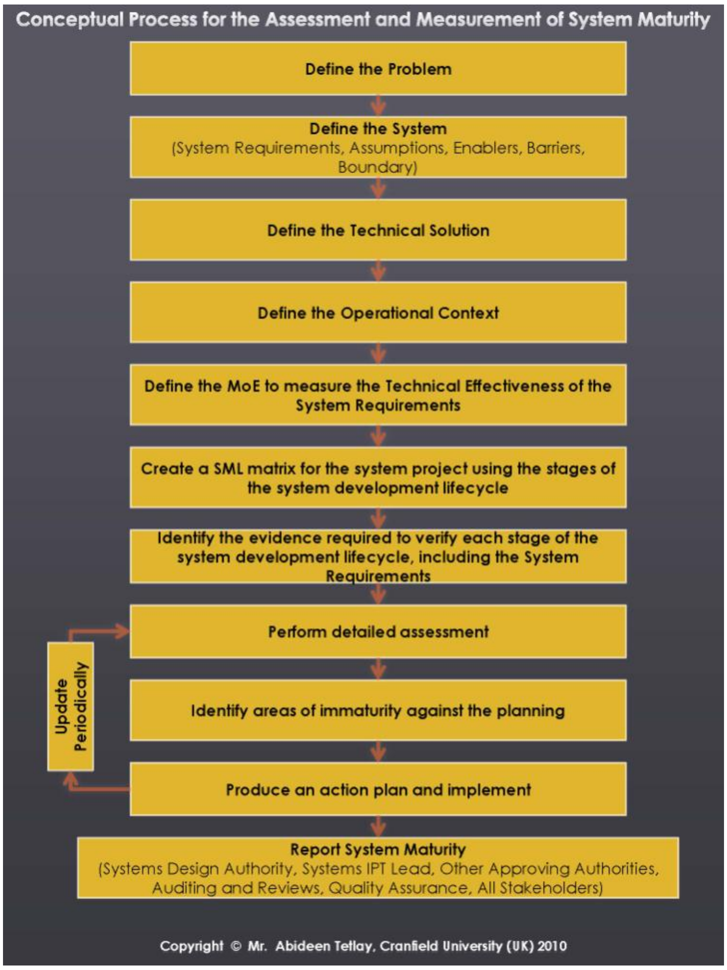


Figure 9: Conceptual Process for the Assessment and Measurement of System Maturity

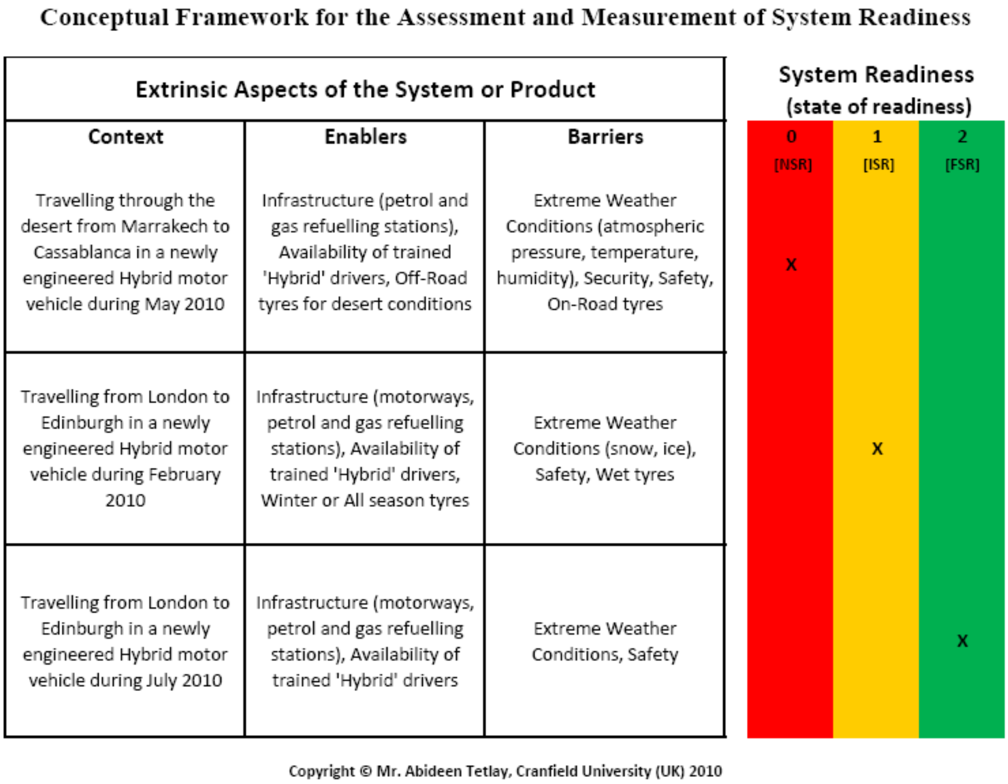


Figure 10: Conceptual Framework for the Assessment and Measurement of System Readiness

An assessment is made for each operational context for the produced-engineered system taking into account the enablers and barriers currently in existence for that context. The degree of System Readiness is then determined and assessed as either achieving: ‘No System Readiness’ (NSR); ‘Initial System Readiness’ (ISR); or ‘Full System Readiness’ (FSR) and conceptually, the state of readiness can be thought of as being equal to either: ‘0’; ‘1’; or ‘2’, respectively. Detailed definitions of these terms are provided in (Tetlay and John 2010) and in this paper under the section heading of ‘System Maturity and System Readiness’. We have provided three examples illustrating the three states of readiness in Figure 10. For simplicity, on this occasion, we have deliberately chosen three non-military defence examples just to illustrate our point in layman’s terms.

The procedure for the assessment and measurement of System Readiness is depicted in Figure 11. It is important to note that you first need to perform a System Maturity assessment and actually achieve System Maturity, i.e. the physical system must physically exist and be developed based on best practice procedures and standards in place and be fully mature and tested as mentioned in the study by (Tetlay and John 2010), before a System Readiness assessment can take place. The main aim of this procedure is to *validate* the User Requirements for a system being engineered in order to achieve System Readiness.

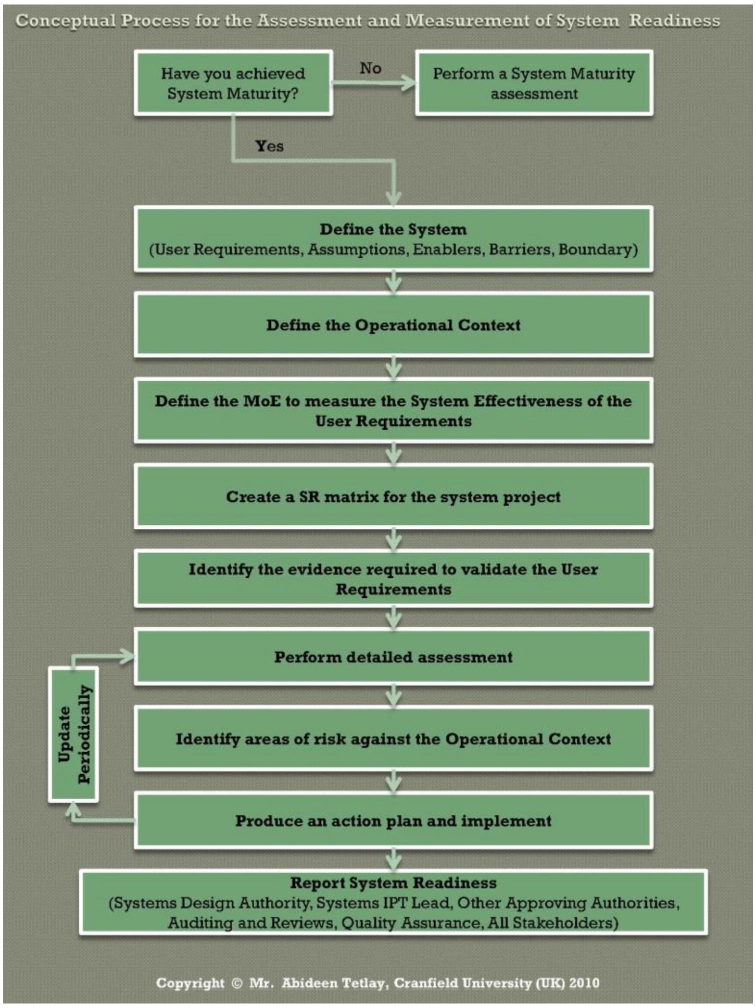


Figure 11: Conceptual Process for the Assessment and Measurement of System Readiness

The conceptual Framework for the assessment and measurement of Capability Readiness is illustrated in Figure 12. Capability Readiness is concerned with the extrinsic aspects of the system with respect to how the system is expected to behave in a ‘total-wider’ system for a particular context subject to certain DLoD and enablers and barriers in place, rather than focusing only on the behaviour of the produced-engineered system for a particular context which is System Readiness.

An assessment is made for each operational context for the system in question taking into account the DLoD and the enablers and barriers currently in existence for that context. An assessment for Capability Readiness will always take into account the DLoD in order to identify and mitigate risks across the DLoD. This was one of the key findings from the case studies: “...risks remain to be overcome on a number of Lines of Development before the Apache can be successfully introduced into service” (NAO 2002).

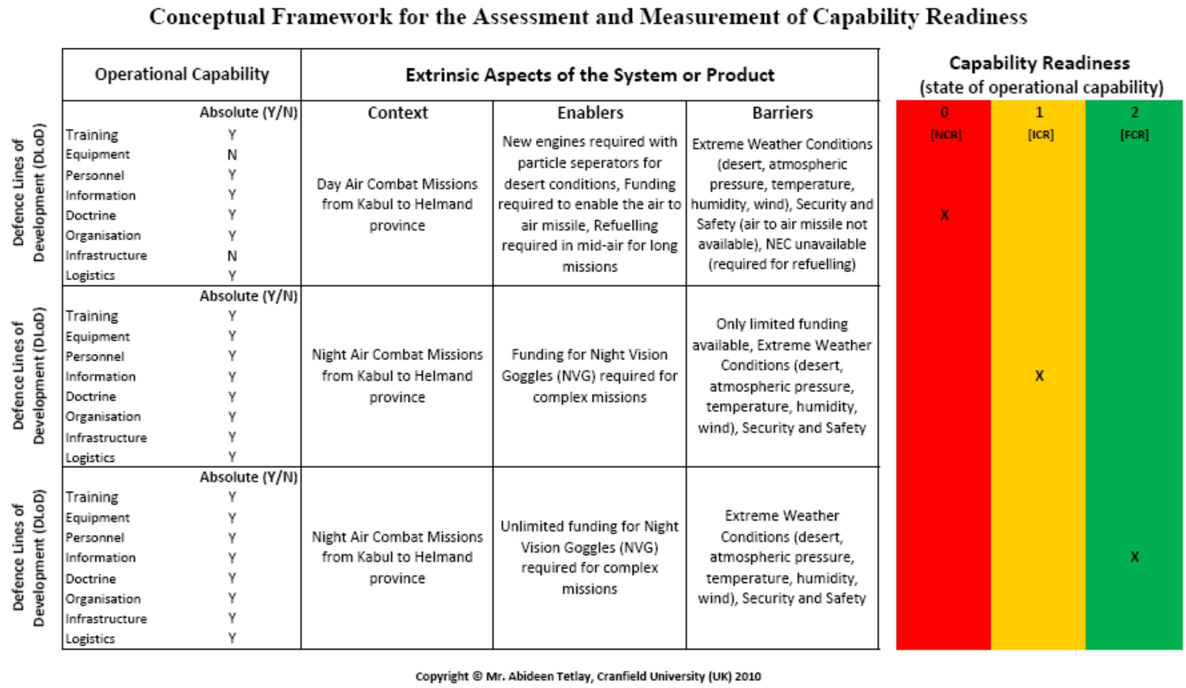


Figure 12: Conceptual Framework for the Assessment and Measurement of Capability Readiness

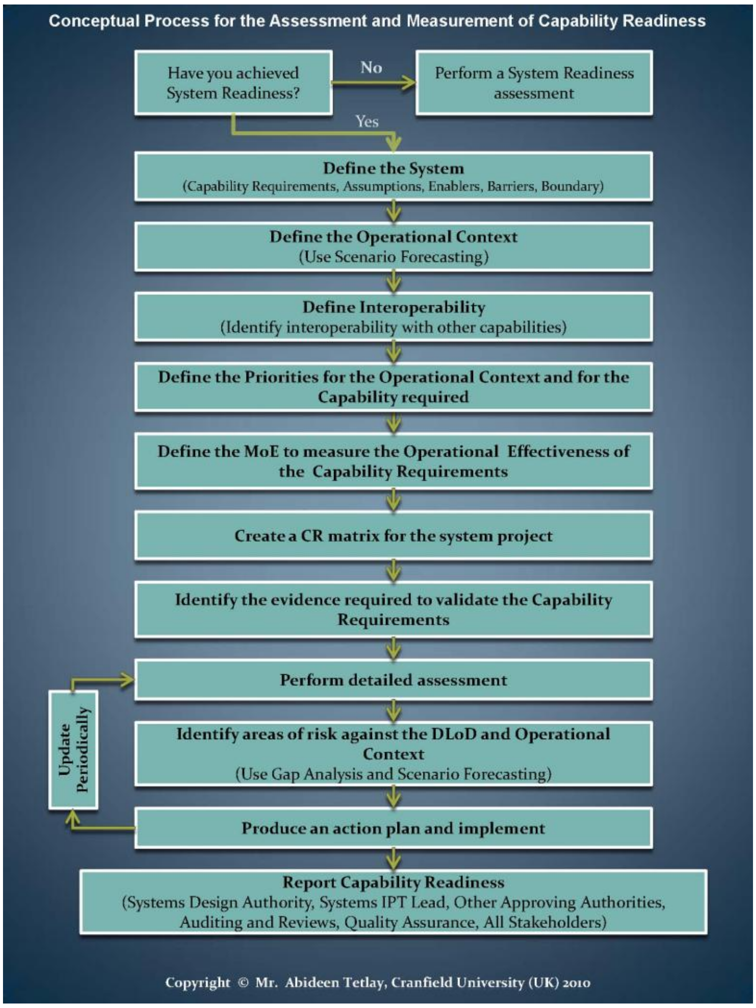


Figure 13: Conceptual Process for the Assessment and Measurement of Capability Readiness

The degree of Capability Readiness is then determined and assessed as either achieving: 'No Capability Readiness' (NCR); 'Initial Capability Readiness' (ICR); or 'Full Capability Readiness' (FCR) and conceptually, the state of operational capability readiness can be thought of as being equal to either: '0'; '1'; or '2', respectively. Detailed definitions of these terms are provided in (Tetlay and John 2010) and in this paper under the section heading of 'System Maturity and System Readiness'. We have provided three non-military defence examples illustrating the three states of operational capability readiness in Figure 12.

The procedure for the assessment and measurement of Capability Readiness is depicted in Figure 13. It is important to note that you first need to perform a System Readiness assessment and actually achieve System Readiness, i.e. the physical system must physically exist and be "ready" for use for a particular context, i.e. the 'Fitness for Purpose' question. Capability Readiness *extends* this notion of readiness and asks the question: Is the system ready for use as a part of a 'total-wider' system? The main aim of this procedure is to *validate* the Capability Requirements for a system operating in a 'total-wider' system in order to achieve Capability Readiness.

## Conclusions

This study summarises the issues with the notions of 'maturity' and 'readiness' within the context of the System Development and overall Lifecycle and provides formal definitions for these terms. It explains why a *Framework* is required to assess and measure Maturity and Readiness and how the conceptual Framework was devised, including a process for applying the Framework for the assessment and measurement of System Maturity, System Readiness and Capability Readiness.

## Further Research

Further research is required to verify the conceptual Framework through refinement and extension with more detail, as appropriate, for wider use and applicability. Alternative perspectives for Framework development, such as Through Life Capability Management and Product-Service Systems (PSS) perspectives using Architectural Frameworks will be considered. This is part of the on-going research.

## Acknowledgements

The author would like to thank the UK Engineering and Physical Sciences Research Council (EPSRC) and BAE Systems for jointly funding the research project.

## References

Boeing (2008), *CH-47D Chinook Helicopter Backgrounder*, available at:
http://www.boeing.com/rotorcraft/military/ch47d/docs/CH-47D_overview.pdf (accessed 7th September 2009)

Boeing (2008), *U.S. Army Special Operations Command Boeing MH-47G Special Operations Chinook Backgrounder*, available at: http://www.boeing.com/rotorcraft/military/ch47d/docs/MH-47G_overview.pdf (accessed 7th September 2009)

CMMISM for Systems Engineering/Software Engineering, (November 2008), Version 1.02 (CMMI-SE/SW, V1.02), Software Engineering Institute, Carnegie Mellon University, available at: http://www.sei.cmu.edu/reports/00tr018.pdf (accessed 10th October 2009)

Downs, D. S. (2007), *Type 45: Design for Supportability*, The Royal Institution of Naval Architects (RINA) International Conference Warship: The Affordable Warship, June 2007, Bath, UK, p. 23-32

FAS (1997), *Operational Requirements Document for the Improved Cargo Helicopter (CH-47D Service Life Extension Program (SLEP))*, Federation of American Scientists (FAS), available at: http://www.fas.org/man/dod-101/sys/ac/docs/ord_ich_jul97.htm (accessed 7th September 2009)

FAS (1999), Federation of American Scientists (1999), *Army returns some Chinooks to flight*, available at:
http://www.fas.org/man/dod-101/sys/ac/docs/a19990827chinback.htm (accessed 7th September 2009)

FAS (2008), Federation of American Scientists (2008), *CH-47 Chinook*, available at:
http://www.fas.org/programs/ssp/man/uswpns/air/rotary/ch47chinook.html (accessed 7th September 2009)

GAO (2001), *Defense Logistics: Information on Apache helicopter Support and Readiness*, Report Number GAO-01-630, United States General Accounting Office (GAO), Report to Congressional Committees

MoD (2008), *What is Through Life Capability Management?* Acquisition Operating Framework (AOF), The high level principles and information that guide how the UK MOD Defence Acquisition Community work version 2.0.6, available at:
http://www.aof.mod.uk/aofcontent/operational/business/capabilitymanagement/capabilitymanagement_whatis.htm

(accessed 10th October 2009)

MoD (2008), *Defence Lines of Development*, Acquisition Operating Framework (AOF), The high level principles and information that guide how the UK MOD Defence Acquisition Community works version 2.0.8, available at: http://aof.mod.uk/aofcontent/strategic/guide/sg_dlod.htm (accessed 10th October 2009)

NAO (2002), *Building an Air Manoeuvre Capability: The Introduction of the Apache Helicopter*, National Audit Office (NAO), Report by the Comptroller and Auditor General, HC 1246, Session 2001-2002, London, UK

NAO (2008), *Ministry of Defence Chinook Mk3 Helicopters*, National Audit Office (NAO), Report by the Comptroller and Auditor General, HC 512, Session 2007-2008, London, UK

NAO (2009), *Ministry of Defence Providing Anti-Air Warfare Capability: the Type 45 Destroyer*, National Audit Office (NAO), Report by the Comptroller and Auditor General, HC 295, Session 2008-2009, London, UK

NAO (2009), *Providing Anti-Air Warfare Capability: the Type 45 Destroyer*, National Audit Office (NAO) Value for Money Report, HC 295 2008-2009, available at: http://www.nao.org.uk/publications/0809/the_type_45_destroyer.aspx (accessed 24th September 2009)

NRI (2009), naval-technology.com, Net Resources International (NRI), available at: http://www.naval-technology.com/projects/horizon/ (accessed 24th September 2009)

NRI (2009), Net Resources International (2009), *Type 45 Daring Class Anti-Air Warfare Destroyers, United Kingdom*, available at: http://www.naval-technology.com/projects/horizon/ (accessed 27th September 2009)

Overkleeft (1996), *TADS/PNVS – The Eyes of the Apache*, Royal Aeronautical Society Conference, 23rd April 1996, Orlando, Florida, US, p. 6.1-6.9

Robson, C. (2002), *Real World Research* (2nd ed), Blackwell Publishing, MA, US

Scott (2006), "UK Royal Navy's Type 45 Destroyer gets ready for launch milestone", *Jane's International Defense Review*, Vol. 39, p. 46-50

Slade (2008), "Apache at 50,000 hours – airpower savagery untamed", *Defence Equipment and Support*, Issue 02, June 2008, p. 18-19

Tetlay, A. and John, P. (2009), *Capability Readiness for Product-Service Systems*, 7th International Conference on Manufacturing Research (ICMR09), September 8th – 10th 2009, Warwick University (UK)

Tetlay, A. and John, P. (2009), *Determining the Lines of System Maturity, System Readiness and Capability Readiness in the System Development Lifecycle*, 7th Annual Conference on Systems Engineering Research (CSER09), April 20th – 23rd 2009, Loughborough University (UK), ISBN: 978-0-9562440-0-0

Tetlay, A. and John, P. (2010), *Clarifying the Concepts of System Maturity, System Readiness and Capability Readiness through Case Studies*, 8th Annual Conference on Systems Engineering Research (CSER 2010), March 17th – 19th 2010, Hoboken, New Jersey, United States

Verma, D. et al., (2003), "System Training Metrics and Measures: A Key Operational Effectiveness Imperative", *Systems Engineering*, Vol. 6, No. 4, p. 238-248

Yin, R. K. (2009), *Case Study Research: Design and Methods* (4th ed), SAGE Publications, California, US

## Biography

**Abideen Tetlay** is currently working as a Systems Engineer for the Raytheon Company based in the UK covering the Systems Engineering Lifecycle to primarily support a major new Command and Control (C2) programme for the UK Ministry of Defence (MoD). He has previously spent 10 years working as a professional Software and Database Engineer reaching Consultant level. He has worked for the following UK based companies: K3 Business Technology Group; Bidwells Property Consultants; Lloyds Banking Group; and the Royal Society of Chemistry. Before embarking on an IT career, he first obtained a Bachelor's degree BA (Hons) Business Administration from the University of Bedfordshire (UK). This was followed by a Master's degree MSc Information Technology (Management Information Systems) from Cranfield University (UK) and then he moved into the IT field. Several years later, he gained another Master's degree MSc Software Engineering from St Cross College, University of Oxford (UK) which was sponsored by Lloyds Banking Group. He is a Fellow of the Institution of Analysts and Programmers (FIAP).